

Amendment to the Claims

Ab

1. (Amended) A method for protecting a sequence computer instructionscode comprising ~~the steps of:~~
preparing first obscuring instructions;
serializing the sequence of computer instructions; and
injecting ~~a large number of~~ second obscuring instructions into the serialized sequence of computer instructionscode in an automated process, using the first obscuring instructions. ~~to produce an obscured sequence of computer instructions that in total is humanly impossible to read and understand; and encrypting a static image of the obscured sequence to protect against direct decompilation.~~

2. (Amended) The method of Claim 1, further comprising ~~the step of executing the serialized sequence of computer instructions injected with the the second obscured instructions, one instruction at a time, thereby making run-time trace and observation a labor intensive manual process.~~

3. (Amended) The method of Claim 1, wherein ~~the obscuring instructions are identified by codes further comprising the steps of:~~ the method further comprises
generating a first set of obscuring instructions having a first set of codes associated therewith;
transforming ~~at~~ the first set of obscuring instruction identification codes associated with some or all of the first set of obscuring instructions to generate into a
second set of obscuring instruction identification codes; and
generating ~~thea~~ second set of obscuring instructions identified by using the
second set of obscuring instruction identification codes.

4. (Amended) The method of Claim 3, wherein the obscuring instruction identification codes comprise ~~are~~ numeric valuescodes and said generating of the
second set of obscuring instruction identification codes comprises performing a

mathematical transformation is performed on the numeric codes values of the first set of obscuring instruction identification codes to produce the numeric codes values of the second set of obscuring instruction identification codes.

5. (Amended) The method of Claim 41, ~~wherein the obscuring instructions are identified by numeric codes further comprising the steps of:~~ further comprising injecting into the serialized sequence of instructions injected with the second obscuring instructions, a description of the mathematical transformation performed.

~~generating a first set of obscuring instructions having a first set of numeric codes associated therewith;~~

~~performing a mathematical transformation on the numeric codes of the first set to produce a second set of numeric codes; and~~

~~generating a second set of obscuring instructions identified by the second set of numeric codes.~~

6. (Amended) The method of Claim 15, ~~further comprising the step of compressing the static image by recording in the static image a record of the transformation used to generate the second set of obscuring instructions.~~ wherein said injecting comprises systematically injecting the second obscuring instructions with a plurality of copies of a runtime manager, forming a plurality of obscured instruction blocks, each comprising a copy of the runtime manager, one or more of the serialized sequence of instructions, and one or more of the second obscuring instructions.

7. (Amended) The method of Claim 61, ~~further comprising the step of compressing the static image.~~ wherein said serialized sequence of instructions, said second obscuring instructions, copies of said runtime manager, and the resulting obscured instruction blocks are in source form, and the method further comprises obscuringly compiling the obscured instruction blocks into object form, preserving the obscuration.

8. (Amended) ~~The method of Claim 74, wherein the obscured sequence of computer instructions is organized into a sequence of blocks of computer instructions and the steps of encrypting a static image comprises the steps of:~~

~~encrypting a first block to form a first encrypted output;~~

~~encrypting a second block and the first encrypted output to form a second encrypted output; and~~

~~encrypting a third block and the second encrypted output to form a third encrypted output.~~

further comprising successively and recursively encrypting up to all, except a root one, of the obscured instruction blocks in object form, to form an obscured executable image having the encrypted ones of the obscured instruction blocks in object form successively nested.

9. (Amended) The method of Claim 8, wherein the successive and recursive encryption further comprises ~~the step of compressing each block the instructions as it is being encrypted.~~

10. (Amended) A method for protecting data of a data file comprising the steps of:

preparing first obscuring data;

injecting a ~~large number of~~ second obscuring data into a plurality of locations in the data file ~~using~~ an automated process and the first obscuring data, to

~~produce~~ organize the data of the data file into an plurality of obscured sequence of data blocks, with each of the obscured data blocks having a portion of the data and one or more obscuring data that in total is humanly impossible to read and understand; and

successively and recursively encrypting a static image of the obscured sequence to protect against direct decompilation the obscured data blocks into a plurality of encrypted obscured data blocks that are successively nested.

11. (Amended) ~~The method of Claim 10, wherein the obscuring data are identified by codes further comprising the steps of: the method further comprises generating a first set of obscuring data having a first set of codes associated therewith;~~

~~transforming the first set of obscuring data identification codes associated with some or all of the first set of obscuring data into to generate a second set of obscuring data identification codes; and~~

~~generating the second set of obscuring data identified by using the second set of obscuring data identification codes.~~

Ab 12. (Amended) ~~The method of Claim 11, wherein the obscuring data identification codes comprise are numeric values ~~codes~~ and said generating of the second set of obscuring data identification codes comprises performing a mathematical transformation is performed on the numeric codes values of the first set of obscuring data identification codes to produce the numeric codes values of the second set of obscuring data identification codes.~~

13. (Amended) ~~The method of Claim 12, wherein the obscuring data are identified by numeric codes further comprising the steps of: further comprising injecting into the data file injected with the second obscuring data, a description of the mathematical transformation performed.~~

~~generating a first set of obscuring data having a first set of numeric codes associated therewith;~~

~~performing a mathematical transformation on the numeric codes of the first set to produce a second set of numeric codes; and~~

~~generating a second set of obscuring data identified by the second set of numeric codes.~~

14. (Cancelled) ~~The method of Claim 13 further comprising the step of comprising the static image by recording in the static image a record of the transformation used to generate the second set of obscuring data.~~

15. (Cancelled) ~~The method of Claim 10 further comprising the step of compressing the static image.~~

16. (Cancelled) ~~The method of Claim 10, wherein the obscured sequence of data is organized into a sequence of blocks of data and the steps of encrypting a static image comprises the steps of:~~

~~encrypting a first block to form a first encrypted output;~~

~~encrypting a second block and the first encrypted output to form a second encrypted output; and~~

~~encrypting a third block and the second encrypted output to form a third encrypted output.~~

a^b
17. (Amended) The method of Claim 106, wherein the successive and recursive encryption further comprises the step of compressing each block as it is the data being encrypted.

18. (Cancelled) ~~The method of Claim 10, further comprising the steps of:~~

~~generating a first set of obscuring data; and~~

~~performing a mathematical transformation on the first set of obscuring data to produce a second set of obscuring data.~~

19. (Amended) Apparatus for protecting a sequence of computer code with obscuring instructions, said apparatus comprising:

means for storing first obscuring instructions;

means for serializing the sequence of computer instructions; and

means for automatically injecting a large number of second obscuring instructions into the sequence of computer code instructions using the first obscuring instructions. to produce an obscured sequence of computer instructions that in total is humanly impossible to read and understand; and

means for encrypting a static image of the obscured sequence to protect against direct decompilation.

20. (Amended) The apparatus of Claim 19, ~~further comprising means for executing the obscuring instructions one instruction at a time, thereby making run time trace and observation a labor intensive manual process wherein the means of injecting are adapted to systematically inject the second obscuring instructions and copies of a runtime manager into the serialized sequence of instructions to form a plurality of obscured instruction blocks.~~

21. (Amended) Apparatus for protecting a sequence of computer code with ~~obscuring instructions comprising:~~

an obscuring instruction bank, ~~for storing to store~~ obscuring instructions each of which is identified by an obscuring instruction identification code;

a transformation function bank ~~for storing to store~~ transformation functions adapted to transform obscuring instruction identification codes; and

a generator ~~for generating to generate~~ blocks of obscuring instructions by selecting identification codes of the obscuring instructions from the stored in obscuring instruction bank, and transformation functions from the transformation function bank, and applying said selected transformation functions being used by the generator to transform the selected obscuring instruction identification codes that identify selected obscuring instructions so as and employ the transformed obscuring instruction identification codes to generate other codes that identify other additional obscuring instructions that are then selected by the generator.

22. (Amended) A method for executing a plurality of critical instructions ~~that are obscured by obscuring instructions and stored along with encrypted loading instructions and decryption keys, said method comprising the steps of:~~

loading a first executable instruction block of an executable module, the first executable instruction block having one or more of the critical instructions, and the executable module further having a plurality of nested encrypted executable instruction blocks having the remaining of the critical instructions that were

generated through successive and recursive encryption into a first memory address a first block of loading instructions, and

executing the loaded first executable instructions block to allocate a first dynamic memory address, including loading the plurality of nested encrypted executable instruction blocks having a first remainder of the critical instructions, retrieving a first decryption key from the loaded plurality of nested encrypted executable instruction blocks, decrypting the loaded plurality of nested encrypted executable instruction blocks once to recover a second executable instruction block and a first remainder of the plurality of nested encrypted executable instruction blocks having a second remainder of the critical instructions. ;

loading and executing at the first dynamic memory address at least one critical instruction that is obscured by obscuring instructions;

retrieving a first decryption key;

decrypting a second block of loading instructions using the decryption key; loading into said first memory address the second block of loading instructions;

executing the loading instructions to allocate a second dynamic memory address; and

loading and executing at the second dynamic memory address at least one more critical instruction that is obscured by obscuring instructions.

23. (Amended) The method of Claim 22, further comprising the steps of:

executing the second executable instruction block, including retrieving a second decryption key from the first remainder of the plurality of nested encrypted executable instruction blocks, decrypting the first remainder of the plurality of nested encrypted executable instruction blocks once to recover a third executable instruction block and a second remainder of the plurality of nested encrypted executable instruction blocks having a third remainder of the critical instructions. retrieving a second decryption key;

decrypting a third block of loading instructions using the second decryption key;

~~loading into said first memory address the third block of loading instructions;~~
~~executing the loading instructions to allocate a third dynamic memory address;~~
~~and~~
~~loading and executing at the third dynamic memory address at least one more critical instruction that is obscured by obscuring instructions.~~

a^b
24. (New) The apparatus of claim 20, wherein the obscured instructions blocks are in source form, and the apparatus further comprises obscuration compiler means for compiling the obscured instruction blocks into a plurality of obscured instruction blocks in object form, preserving the obscuration.

25. (New) The apparatus of claim 24, wherein the apparatus further comprises encryption means for successively and recursively encrypting up to all, less a root one, of the obscured instruction blocks in object form, to form an obscured executable image having the encrypted ones of the obscured instruction blocks in object form successively nested.

26. (New) The apparatus of claim 25, wherein the encryption means includes compression means for compressing instructions being encrypted.

27. (New) The apparatus of claim 21, further comprising an injector to automatically inject the additional obscuring instructions and copies of a runtime manager into the sequence of computer instructions, to form a plurality of obscured instructions blocks, each comprising a copy of the runtime manager, one or more of the computer instructions, and one or more of the obscuring instructions.